

Image Steganography Project Report

Submitted

by

M. George Steven - AP19110010151

G. Nishita - AP19110010302

D. Sai Praveen - AP19110010215

J. Raghavendra - AP19110010222

P. Surya Teja- AP19110010299

B. RajKalyan-AP19110010194

Under the guidance of Dr. Ashok Kumar Pradhan



Department of Computer Science and Engineering

SRM University-AP, Andhra Pradesh, India

TABLE OF CONTENTS

Sl. No.	Content	Page No.
1	Abstract	3
2	Introduction	3
3	Literature Survey	4
4	Proposed Model	4-5
5	Contributions	5
6	Code	6-8
7	Result	9-12
8	Conclusion	12
9	Reference	12-13

I. ABSTRACT

- Steganography is the art of hiding the fact that communication is taking place by hiding information in other information.
- Many different carrier file formats can be used, but digital images are the most popular because of their frequency on the internet.
- For hiding secret information in images, there exists a large variety of steganography techniques, some are more complex than others and all of them have respective strong and weak points.
- Different applications may require absolute invisibility of the secret information, while others require a large secret message to be hidden.
- This project report intends to give an overview of image steganography, its uses, and techniques.
- **Objective:** The objective is to send data securely between two parties without being suspected by anyone that it has data hidden in it.

II. INTRODUCTION

- **Terms & Terminologies used:** We used python to design a simple application interface for the ease of users to use it. And we used the “stegano” library to get the job done to hide the data in the image.
- **Motivation:** Our motive was to promote safer and simpler transfer of data between two persons or parties using images so that no one knows that data is being sent between them except those two.
- **Problem statement:**
 - ❖ How can we send a message secretly to the destination?
 - ❖ Using steganography, information can be hidden in carriers such as images, audio files, and text files.
 - ❖ In this study, we proposed a new framework of an image steganography system to hide the digital text of a secret message.

III. LITERATURE SURVEY

We can see in the daily world, how the internet is being used. At which high levels we can use, share & receive data from anywhere in the world. This has been an advantage and also a huge problem as days go by. Many people & researchers have been working on reducing the threat level and increasing the security measures. As part of this, a solution is using the utility or the benefits of Steganography & Cryptography.

The word 'Steganography' means it is covered writing that is derived from the Greek language. There are a lot of studies and proposed methods that can combine these both to create one proper system. Moreover, this project had been accomplished based on the requirements of security and confidentiality. If we go on and see we can find the number of threats increasing goes on the process of solving threats. From this, we can find only that using steganography will be the best approach & the technique to avoid further threats.

So, we have proposed an encode & decode technique by using the Steganography method to hide the data in the images. In Addition, we have used an algorithm & the steganographic technique, so one can make the communication confidently, safe & securely. The secret message will be finally embedded and delivered secretly. The receiver's end can retrieve the secret message through the decoding process.

IV. PROPOSED MODEL

Algorithm:

- The first thing that appears at the start of an application is where the user can choose the encode option or decode option.
- Later, if the user chooses to select the option as encode.
- Then it opens a new application, where the person can give the secret message which needs to be encrypted and will be given as the input.
- Moreover, we can choose the file where we want to save the file by choosing the file name for it.
- Later we need to select & upload the image from our desktop as we are hiding the message in the image.
- So, now after filling in the details and selecting the image we can just encode it by selecting the encode button.

- We will get a popup dialogue box, after which it says whether do you want to continue to encode or not which gives two options as yes or no.
- If we continue to encode click “yes” or click “no”, where the changes won’t be made if clicked “No”.
- So if we wish to proceed to encode in the dialogue box, we will get successfully encoded.
- Now whereas if we wish to decode what we have encoded.
- Then simply choose the Decode option.
- And after we choose the decode, it opens a new application where we can choose the file.
- By selecting the file from our desktop which is saved already. And by clicking the decode.
- The secret message appears as we have decoded it.

V. CONTRIBUTIONS

George:

- Had written the main code required for the project.
- Moreover also helped in error corrections.

Sai Praveen:

- Designed the application for the project.
- Explained the Tkinter module

Nishita:

- Helped in writing the algorithm
- Additionally, helped in writing the report.

RajKalyan:

- Has written most of the report
- Optimised the code to reduce the number of lines

Raghavendra

- Introduced the stegano module to the team members.
- Explained the project so that everyone could understand well

Surya:

- Helped on solving the errors in the code
- Also helped in designing application

VI. CODE

```
●   from tkinter import *
●   from tkinter.filedialog import *
●   from PIL import ImageTk,Image
●   from stegano import exifHeader as stg
●   from tkinter import messagebox
●
●
●   def encode():
●       main.destroy()
●       enc=Tk()
●       enc.title("encode")
●       enc.geometry("500x400+300+150")
●
●
●       label1=Label(text="Secret message")
●       label1.place(relx=0.1,rely=0.1,height=20,width=80)
●
●
●       entry=Entry()
●       entry.place(relx=0.4,rely=0.1)
●
●
●       label2=Label(text="File name")
●       label2.place(relx=0.1,rely=0.2,height=20,width=80)
●
●
●       entrysave=Entry()
●       entrysave.place(relx=0.4,rely=0.2)
●
●
●       def openfile():
●           global fileopen
●           fileopen=StringVar()
●           fileopen=askopenfilename(initialdir="/Desktop",title="Select
●           File",filetypes=((("jpeg files", "*jpg"), ("all files","*.*"))))
●
●           label3=Label(text=fileopen)
●           label3.place(relx=0.3,rely=0.3)
```

```
def encodee():

    response=messagebox.askyesno("pop up","do you want to
encode")

    if response==1:

        stg.hide(fileopen,entrysave.get()+' .jpg',entry.get())

        messagebox.showinfo("pop up","successfully encoded")


    else:

        messagebox.showwarning("pop up", "unsuccessful")


buttonselect=Button(text="Select File",command=openfile)
buttonselect.place(relx=0.1,rely=0.3)

buttonencode=Button(text="Encode",command=encodee)
buttonencode.place(relx=0.4,rely=0.5)

def decode():

    main.destroy()

    dnc=Tk()

    dnc.title("decode")
    dnc.geometry("500x400+300+150")

    def openfile():

        global fileopen

        fileopen=StringVar()

        fileopen=askopenfilename(initialdir="/Desktop",title="Select
File",filetypes=((("jpeg files", "*jpg"), ("all files","*.*"))))

    def decodee():

        message=stg.reveal(fileopen)
```

```
●     label4=Label(text=message)
●     label4.place(relx=0.3,rely=0.3)

●
●
●     buttonselect=Button(text="Select File",command=openfile)
●     buttonselect.place(relx=0.1,rely=0.3)

●
●     buttonencode=Button(text="Decode",command=decodeee)
●     buttonencode.place(relx=0.4,rely=0.5)

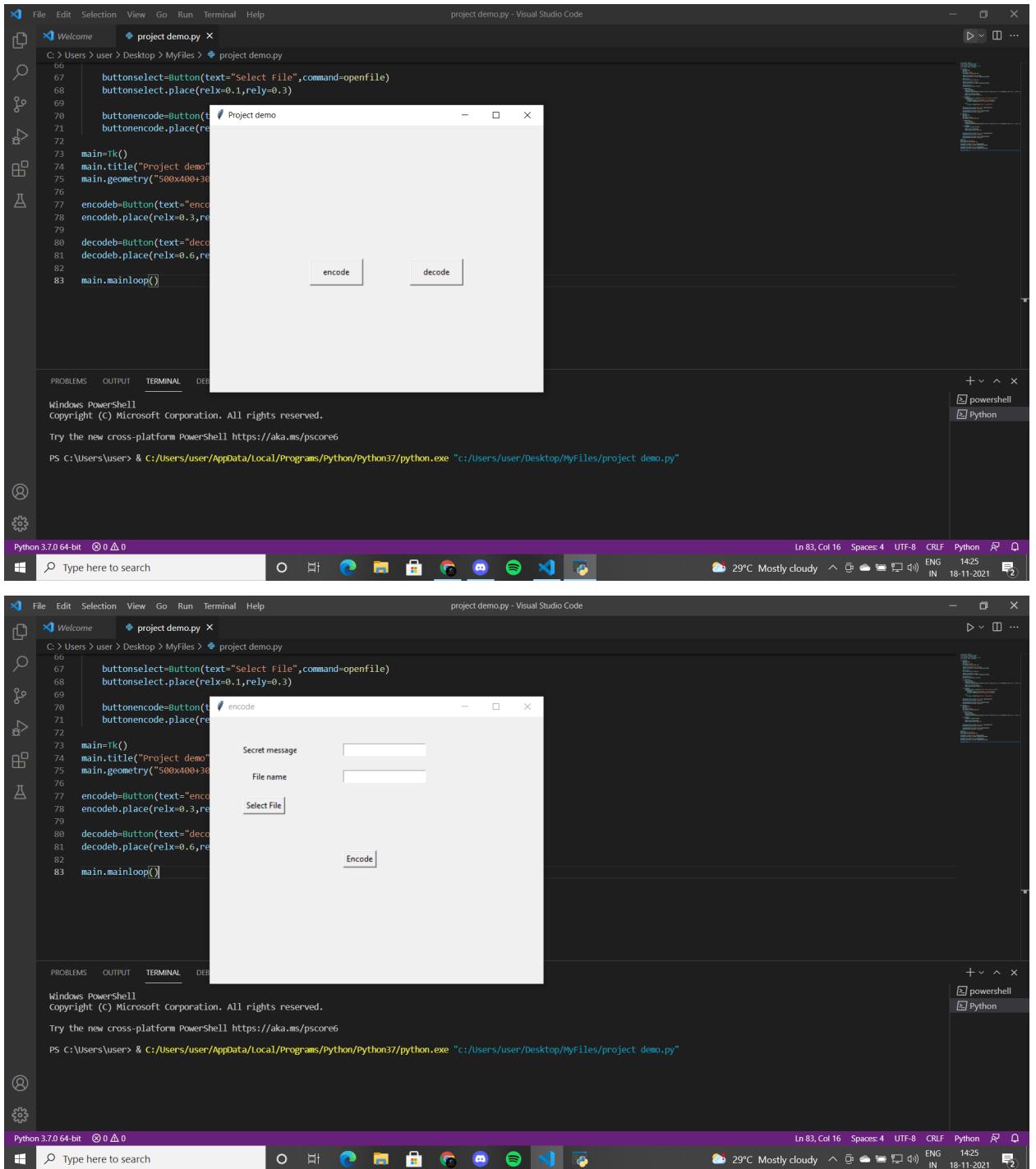
●
●     main=Tk()
●     main.title("Project demo")
●     main.geometry("500x400+300+150")

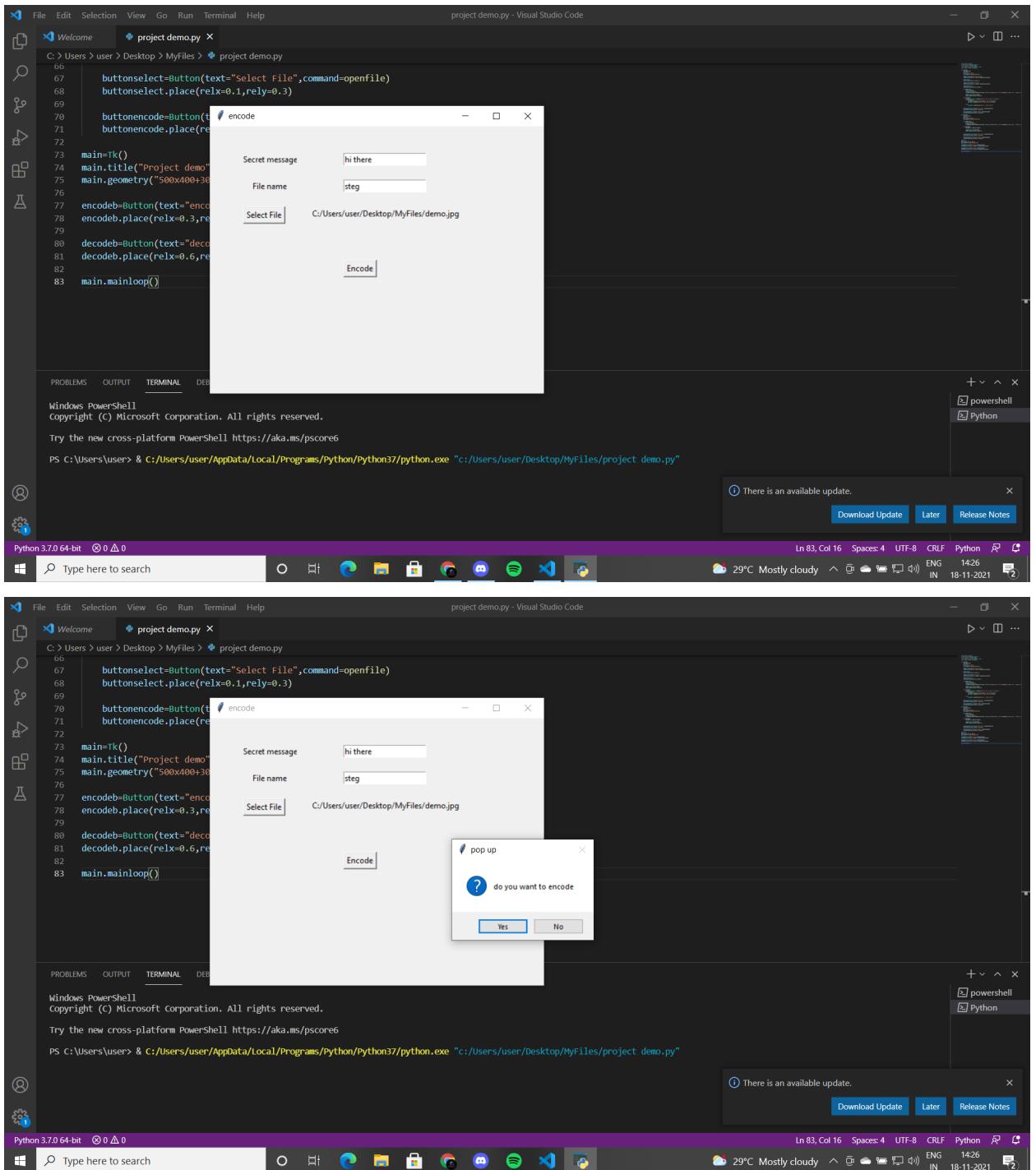
●
●     encodeb=Button(text="encode",command=encode)
●     encodeb.place(relx=0.3,rely=0.5,height=40,width=80)

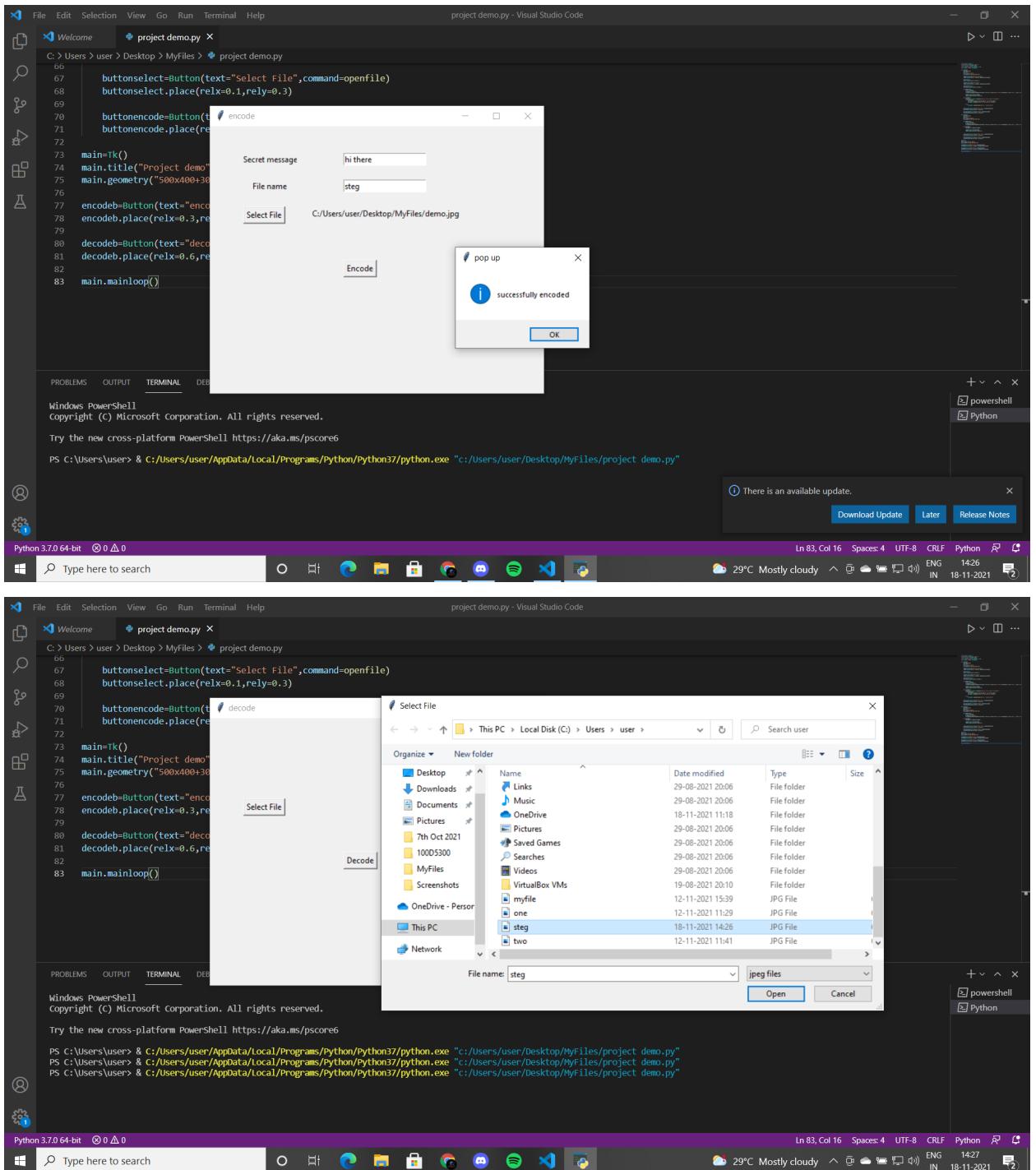
●
●     decodeb=Button(text="decode",command=decode)
●     decodeb.place(relx=0.6,rely=0.5,height=40,width=80)

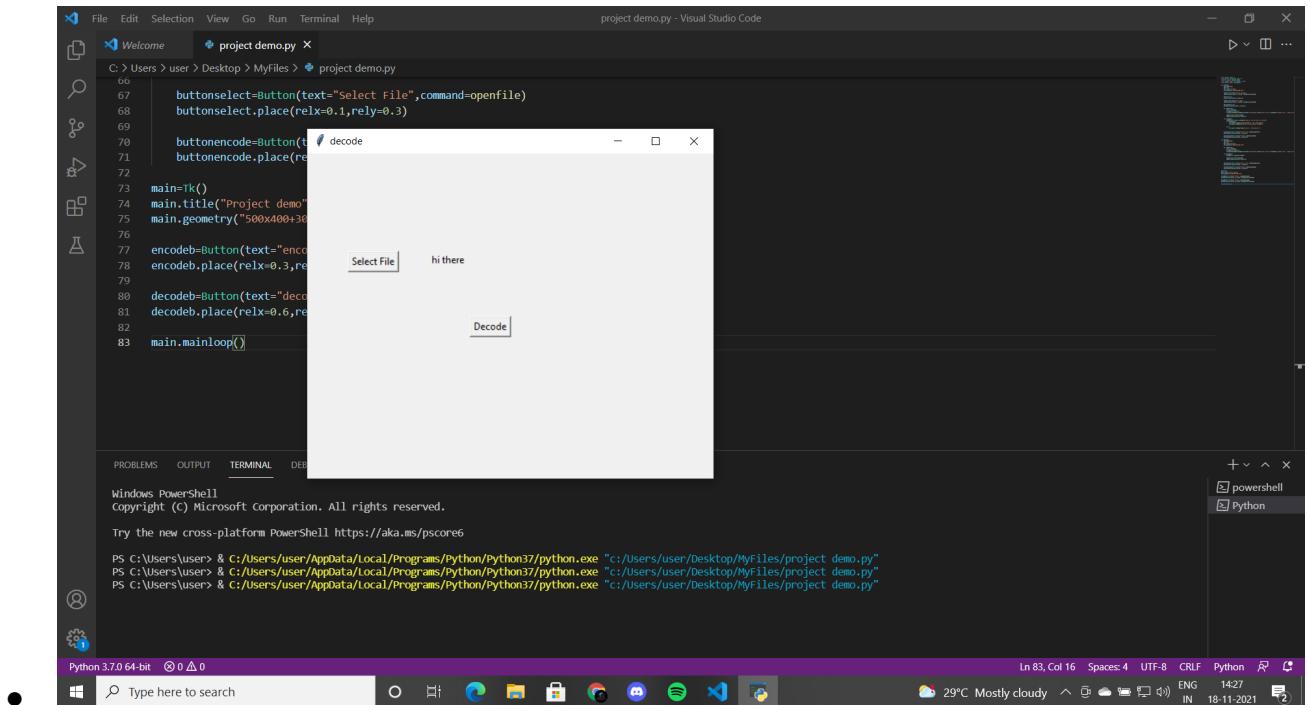
●
●     main.mainloop()
```

VII. RESULTS/OUTPUT:









VIII. CONCLUSION

→ In our project we have designed the steganography to make a greater performance. As the secret message is hidden inside the image the attacker never knows that data is hidden inside the image. We have implemented our project with python programming and also with the help of a few libraries. The method proposed here has successfully been hiding various messages in images and transferring them. This can conclude that this can transfer the information securely between the two parties aside from sender & receiver end. In addition, the proposed method is easy & simply can be implemented. By the results, we can also confirm that we have prevented from attacker detecting the date hidden.

IX. REFERENCES

- <https://towardsdatascience.com/steganography-hiding-an-image-inside-another-77ca66b2acb1>
- https://www.researchgate.net/publication/258789911_A_Steganography_Technique_for_Hiding_Image_in_an_Image_using LSB_Method_for_24_Bit_Color_Image

➤ <http://cse.anits.edu.in/projects/projects1920A17.pdf>

